Penelope: The NBTI-Aware Processor

Xavier Vera

Jaume Abella, Antonio Gonzalez

CTG/MTL Intel Barcelona Research Center – Intel Labs UPC

3/12/2007

MICRO 40, Chicago

Purpose

NBTI is one of the main sources of failure affecting transistors

- NBTI affects PMOS transistors when voltage at the gate is negative
- PBTI (affecting NMOS) is gaining importance
- FMAX and Vmin are impacted
- NBTI can be mitigated controlling different parameters, such as operating voltage, temperature, geometry and duty cycle

We propose a set of microarchitectural mechanisms to manage inputs and contents of blocks so that duty cycle of PMOS is lowered

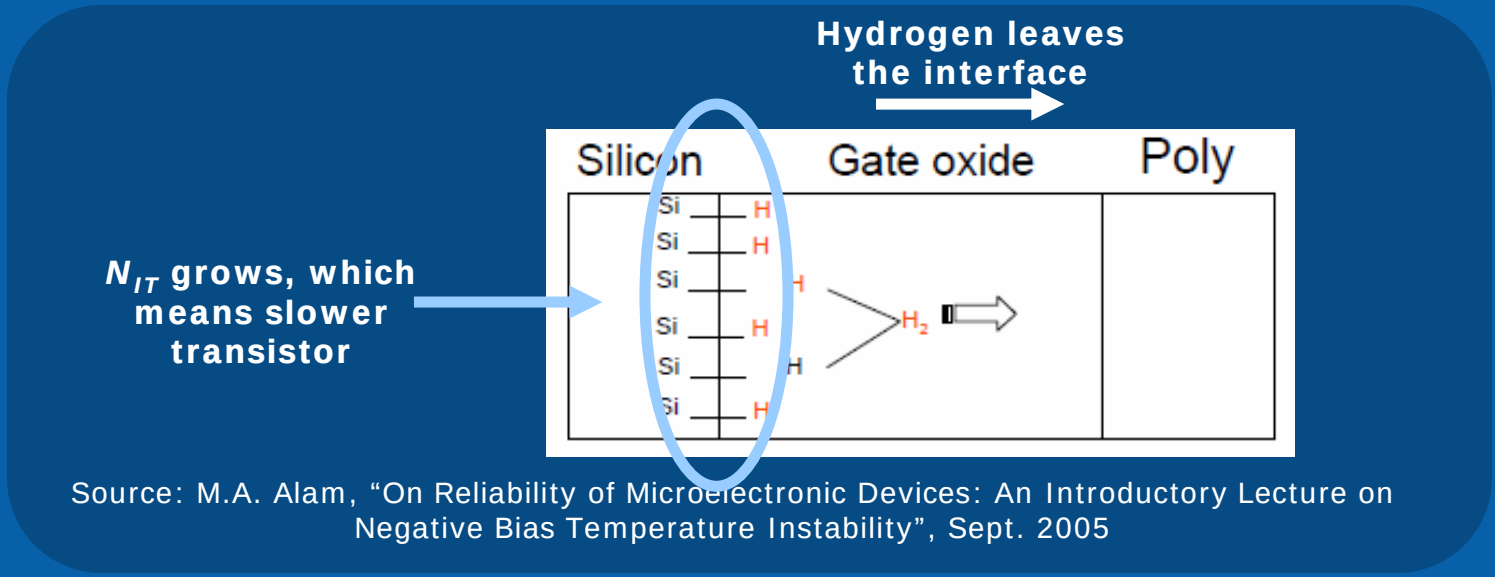
Agenda

- Understanding NBTI and its impact
- Solutions for sequential blocks
- Solutions for combinational blocks
- Conclusion

NBTI Degradation

NBTI affects PMOS transistors when voltage at the gate is negative: Si-H bonds break

- More traps (N_{IT}) in the interface make the transistor slower

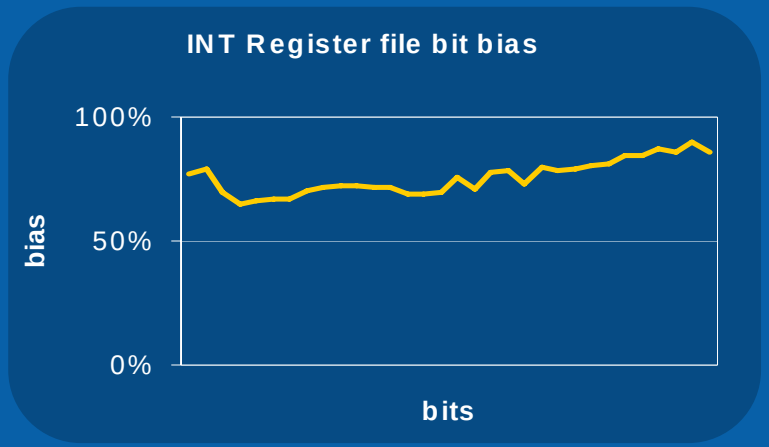


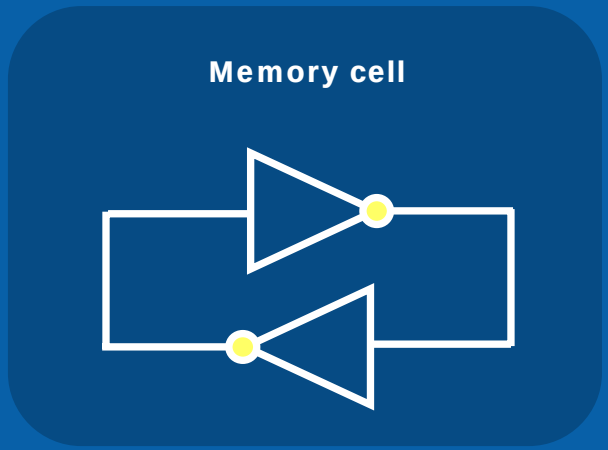
Source: M.A. Alam, "On Reliability of Microelectronic Devices: An Introductory Lecture on Negative Bias Temperature Instability", Sept. 2005

Impact of NBTI

PMOS transistors degrade only when they have a “0” at their gates

- High bias of data
- Memory cells suffer higher bias because of their design
- Worst bit determines lifetime





NBTI: Current Solutions

Current solution is guardbanding

- Vmin is increased ~10%. Higher power dissipation because of higher Vmin
- FMAX is reduced ~10-20%

Sources:

- W. Abadeer, W. Ellis. Behavior of NBTI under AC Dynamic Circuit Conditions. In IRPS 2003.

- M. Agostinelli et al. Erratic Fluctuations of SRAM Cache Vmin at the 90nm Process Technology Node. In IEDM 2005.

Mitigating the NBTI Problem

NBTI can be mitigated controlling different parameters

- Operating voltage: lower voltage means lower NBTI
- Temperature: lower temperature means lower NBTI
- Geometry: wider transistors suffer lower NBTI
- Duty cycle: lower fraction of time with “0” at the gate means lower NBTI

Voltage, temperature and geometry impact delay, power and area. Duty cycle is easy to manage with microarchitectural techniques

Agenda

- Understanding NBTI and its impact
- Solutions for sequential blocks
- Solutions for combinational blocks
- Conclusion

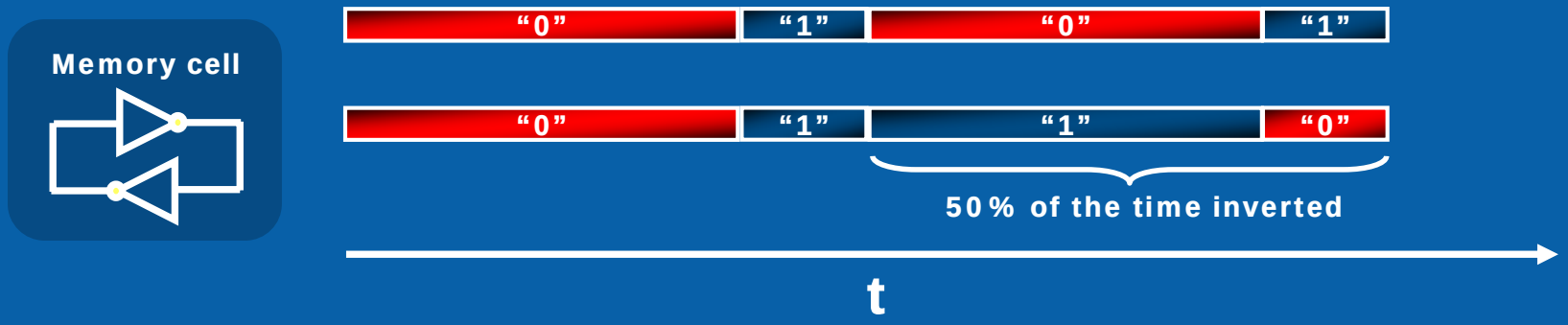
Managing Contents in Storage

Lowest degradation achieved when both PMOS degrade the same

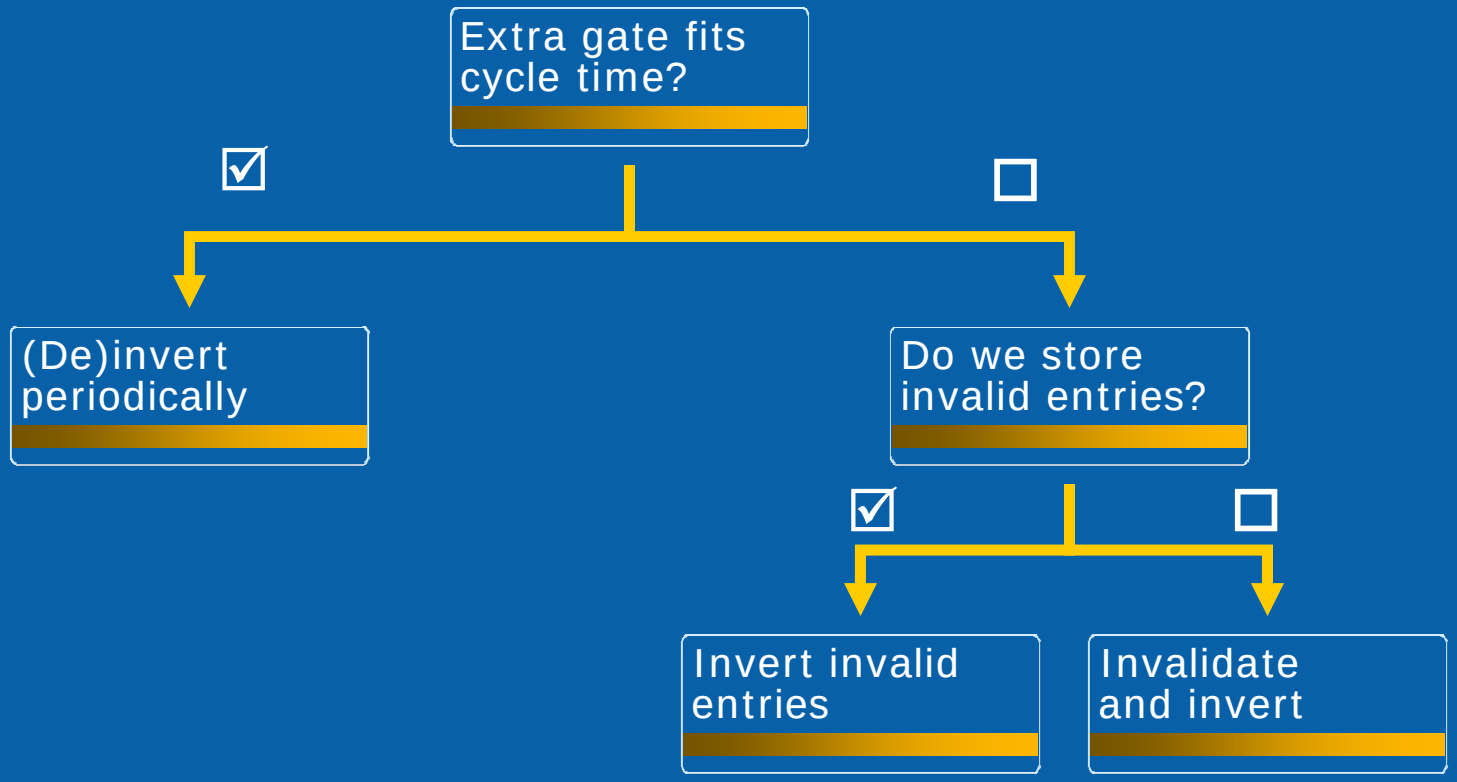
- Hence, we want to store “0” and “1” 50% of the time each

Keep contents inverted 50% of the time in such a way that perfect balancing is achieved

- Data is highly biased



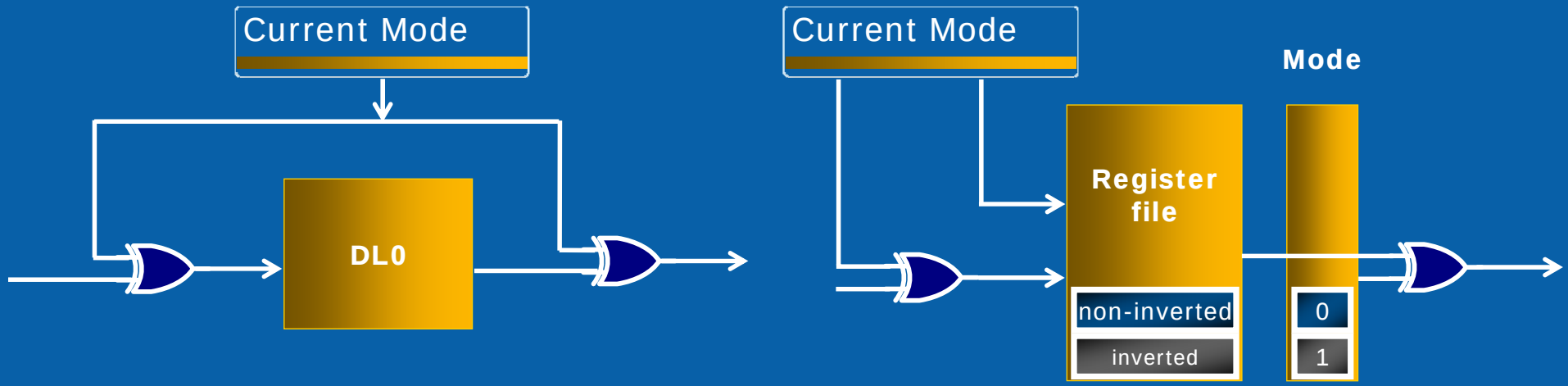
Choosing What to Invert



Case 1: (De)Invert Periodically

Two operating modes: non-inverted and inverted

50% of the time in each mode



Source: Wilkerson et al, CTG/MTL

Case 2: Inverting Invalid Contents

Characteristics:

- 50% of the storage contents are inverted at any time
- In the long run all entries will spend 50% of the time inverted

Low overhead

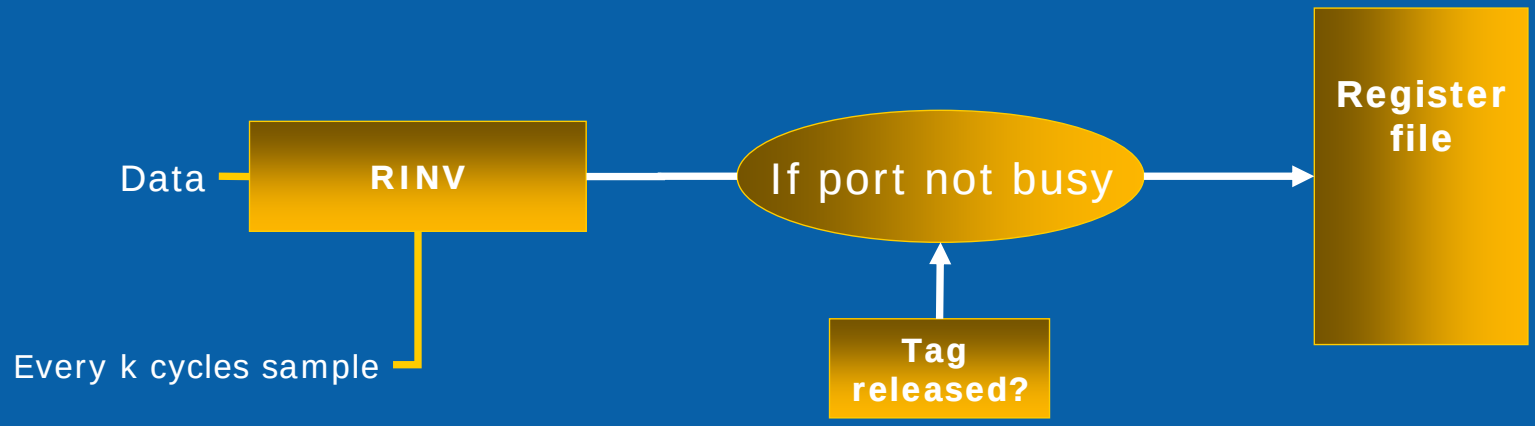
- Actual ports are used (no extra ports required)
- Some extra logic is required (off the critical path)
- Cycle time, TDP and area are roughly the same

Inverting Invalid Contents: Register File

Observation: registers spend more than 50% of the time with invalid contents

- Contents are invalid since they are released until they are written again

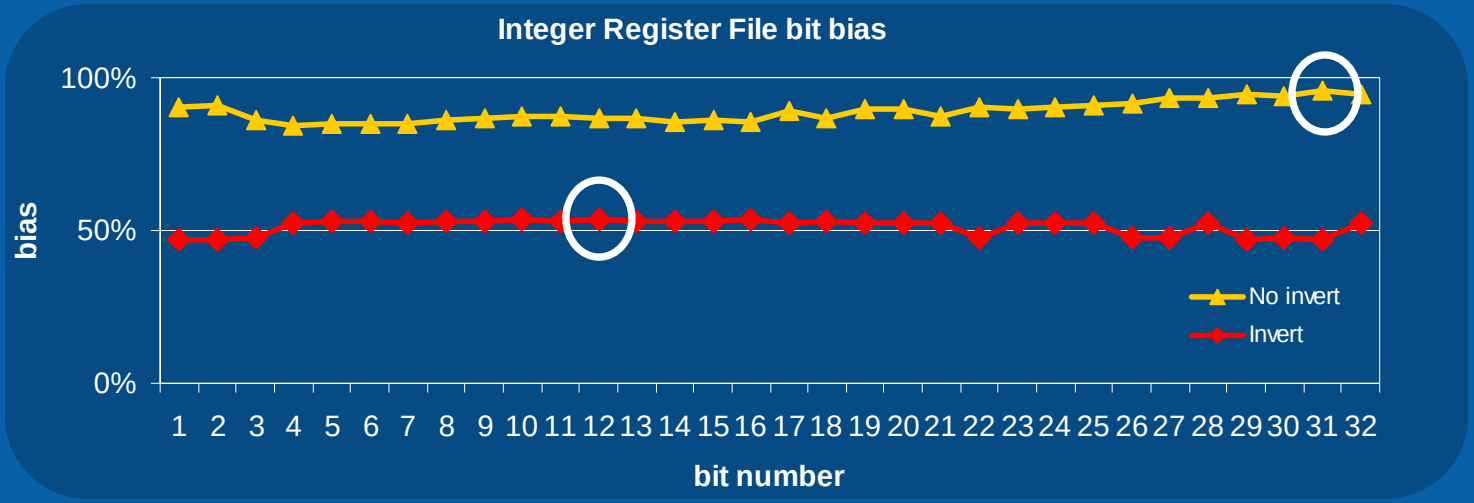
We invert register's contents when they are released



Results for Integer Register File

Guardband in FMAX can be smaller (higher performance) and Vmin can be reduced (lower power)

- Bias without inverting: 96% (46% from the optimal)
- Bias inverting: 53.5% (3.5% from the optimal)



Inverting Invalid Contents: Scheduler

Entries are busy more than 50% of the time

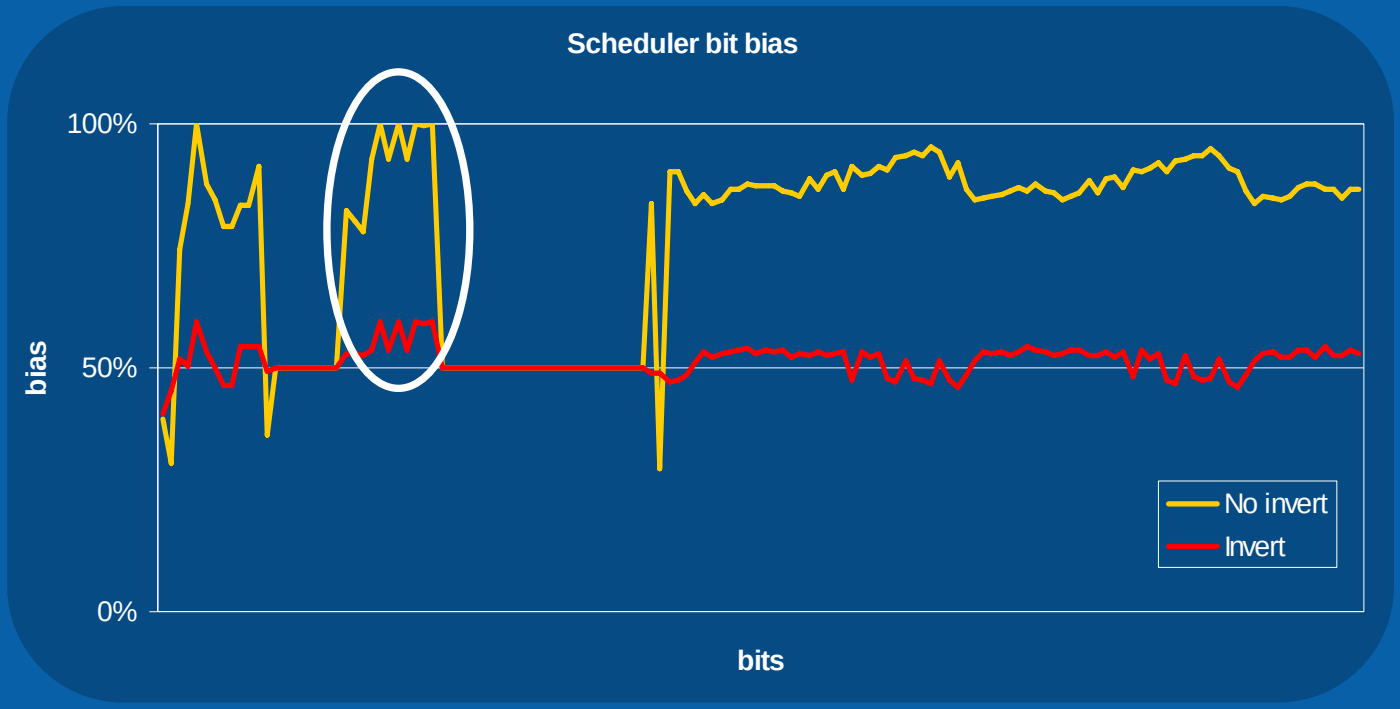
- Some entries are self balanced because they store tags
- Entries that are busy < 50%, like the register file
- Entries busy > 50%, inverting is not enough.
  - If “0” (“1”) most of the time, write “1” (“0”) when idle

Some entries are too biased or too busy and perfect balancing cannot be achieved

- Lower benefits in terms of guardband

Results for Scheduler

Worst bit bias reduces from 100% to 60% (ideally we want 50%)



Case 3: Invalidate & Invert: Cache Structures

HOW/WHAT TO INVERT

Any entry may be useful in caches

- If an entry is inverted, we need to invalidate it

Evict likely-dead entries (most of them are dead)

- Those close to the LRU are unlikely to be reused

WHEN TO INVERT

Keep 50% entries inverted at any time

- Easy to implement
- May lose some performance

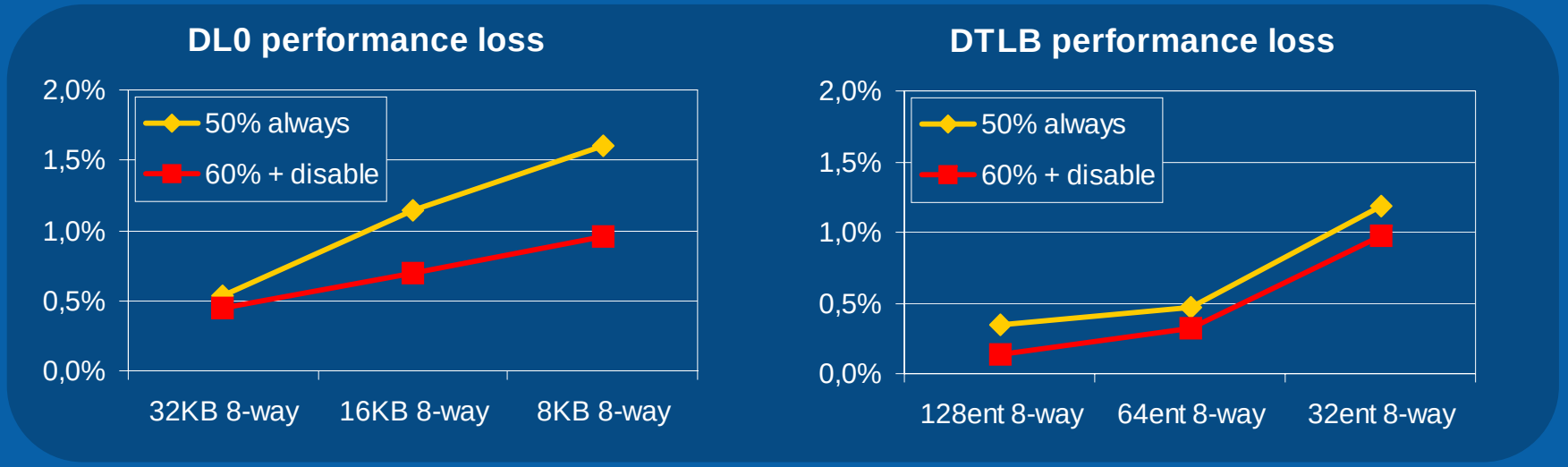
Invert more than 50% when no performance loss is expected, and less than 50% otherwise

- A bit more complex
- Has fewer *glass-jaws*

Results for DL0 and DTLB

Near-optimal balancing is achieved

Benefits in terms of both FMAX and Vmin overcome performance loss (most of the FMAX guardband can be removed)



Agenda

- Understanding NBTI and its impact
- Solutions for sequential blocks
- Solutions for combinational blocks
- Conclusion

Managing Inputs for Combinational Logic

During idle periods inputs remain the same

- The same PMOS degrade

If different inputs are used, different PMOS will degrade

- None of the PMOS will degrade too much
- Maximum duty cycle is reduced

Set special inputs during idle periods to reduce maximum degradation

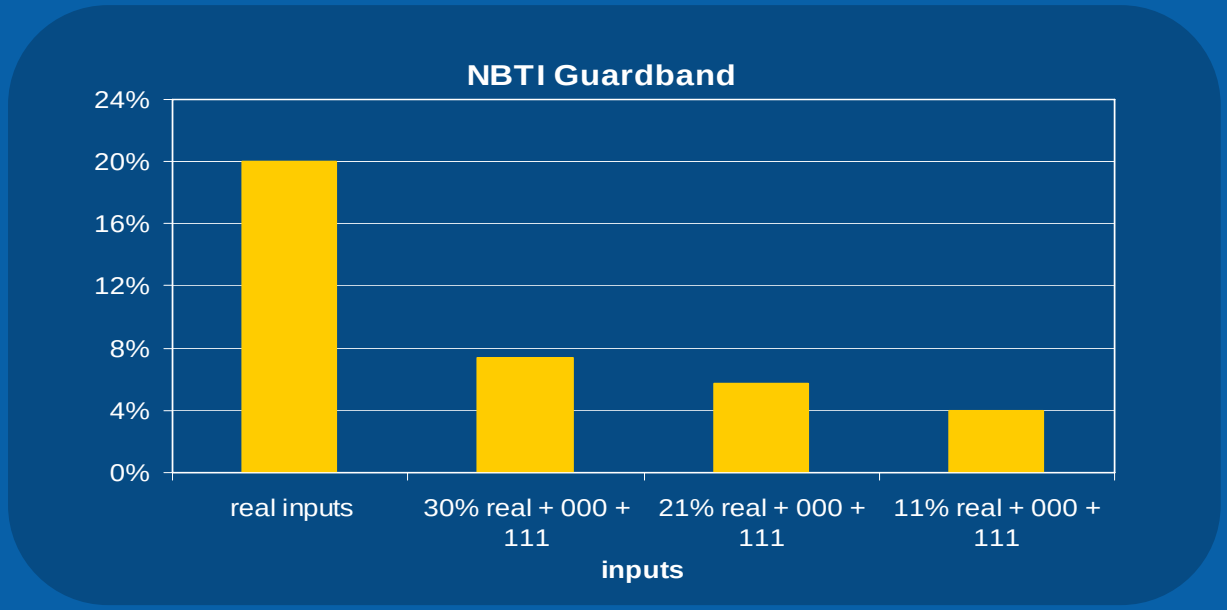
- Alternate different inputs in a round-robin fashion during idle periods

More details in the paper

Results for an Adder

Inputs have been chosen to degrade different PMOS

Few inputs are enough to balance the degradation



Agenda

- Understanding NBTI and its impact
- Solutions for sequential blocks
- Solutions for combinational blocks
- Conclusion

Conclusion

Whole processor protected with minor modifications

Sequential blocks

- Non-critical access time: invert periodically all contents
- Critical access time: invert invalid contents

Combinational blocks

- Set special inputs during idle periods

Guardbanding may be reduced

- FMAX is higher → performance is boosted; Vmin is lower → power is reduced

Future work: take a look to NMOS (PBTI)

Q&A

Acknowledgements

Alex Pineiro, IBRC/MTL

Javier Carretero, IBRC/MTL

Pedro Chaparro, IBRC/MTL

Chris Wilkerson, OML/MTL

Nam Sung Kim, CRL/MTL

Ronny Ronen, IDC/MTL

Shekhar Borkar, MTL

Joe Schutz, MTL